

## Refine Search

### Search Results -

Terms	Documents
L23 and @pd > 20061010	156

**Database:**

US Pre-Grant Publication Full-Text Database  
 US Patents Full-Text Database  
 US OCR Full-Text Database  
 EPO Abstracts Database  
 JPO Abstracts Database  
 Derwent World Patents Index  
 IBM Technical Disclosure Bulletins

**Search:**

L24

[Refine Search](#)

[Recall Text](#)

[Clear](#)

[Interrupt](#)

### Search History

**DATE:** Wednesday, March 14, 2007    [Purge Queries](#)    [Printable Copy](#)    [Create Case](#)

<u>Set</u>	<u>Name</u>	<u>Query</u>	<u>Hit Count</u>	<u>Set</u>
				<u>Name result set</u>
side by side				
DB=PGPB,USPT,USOC,EPAB,JPAB,DWPI,TDBD; PLUR=YES; OP=ADJ				
<a href="#"><u>L24</u></a>	<a href="#"><u>L23</u></a>	L23 and @pd > 20061010	156	<a href="#"><u>L24</u></a>
<a href="#"><u>L23</u></a>		"knowledge base" and (database or "data structure" with component\$) and ("object relational hierarchy" with precedence or node\$)	2344	<a href="#"><u>L23</u></a>
<a href="#"><u>L22</u></a>		L18 and ((higher or lower) adj precedence adj level)	5	<a href="#"><u>L22</u></a>
<a href="#"><u>L21</u></a>		L18 and (higher or lower adj precedence adj level)	1585	<a href="#"><u>L21</u></a>
<a href="#"><u>L20</u></a>		L18 and (lower adj precedence adj level)	0	<a href="#"><u>L20</u></a>
<a href="#"><u>L19</u></a>		L18 and (higher adj precedence adj level)	5	<a href="#"><u>L19</u></a>
<a href="#"><u>L18</u></a>		"knowledge base" and (database or "data structure") and ("object relational hierarchy" with precedence or node\$)	2487	<a href="#"><u>L18</u></a>
<a href="#"><u>L17</u></a>		("knowledge base" and (database or "data structure") and ("object relational hierarchy")) and ("precedence level") or node)	444863	<a href="#"><u>L17</u></a>
<a href="#"><u>L16</u></a>		"knowledge base" and (database or "data structure") and ("object relational hierarchy")	5	<a href="#"><u>L16</u></a>

<u>L15</u>	"knowledge base" and (database or "data structure") and (object same relational same model) and (rule with object and object adj "relational database")	35	<u>L15</u>
<u>L14</u>	"knowledge base" and (database or "data structure") and (object same relational same model)	232	<u>L14</u>
<u>L13</u>	database same (individual or each) adj rule adj object	1	<u>L13</u>
<u>L12</u>	database same individual same rule same object	269	<u>L12</u>
<u>L11</u>	database individual rule object	0	<u>L11</u>
<u>L10</u>	"knowledge base" and (object with relational with (preference adj1 level))	0	<u>L10</u>
<u>L9</u>	715/.ccls. and ("knowledge base" and (object with relational) and "data structure" and (document with generat\$ same content))	5	<u>L9</u>
<u>L8</u>	707/.ccls. and ("knowledge base" and (object with relational) and "data structure" and (document with generat\$ same content))	9	<u>L8</u>
<u>L7</u>	"knowledge base" and (object with relational) and "data structure" and (document with generat\$ same content)	28	<u>L7</u>
<u>L6</u>	"knowledge base" and (object with relational) and "data structure" and (preference with rule\$ near3 content)	0	<u>L6</u>
<u>L5</u>	"knowledge base" and (object with relational) and "data structure"	280	<u>L5</u>
<u>L4</u>	L2 and ("knowledge base" same object) with relational	4	<u>L4</u>
<u>L3</u>	L2 and ("knowledge base" same object)	7	<u>L3</u>
<u>L2</u>	6006242	78	<u>L2</u>
<u>L1</u>	5666130	14	<u>L1</u>

END OF SEARCH HISTORY


[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)
 The ACM Digital Library  The Guide



THE ACM DIGITAL LIBRARY

[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

Terms used **kwnoledge base objects read only precedence precedence level**

Found 51 of 198,617

Sort results by

 relevance 
 [Save results to a Binder](#)
 Try an [Advanced Search](#)

Display results

 expanded form 
 [Search Tips](#)
 Try this search in [The ACM Guide](#)
 Open results in a new window

Results 1 - 20 of 51

Result page: **1** [2](#) [3](#) [next](#)Relevance scale     
**1 [OOPAL: integrating array programming in object-oriented programming](#)**

Philippe Mougin, Stéphane Ducasse

 October 2003 **ACM SIGPLAN Notices , Proceedings of the 18th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications OOPSLA '03**, Volume 38 Issue 11

Publisher: ACM Press

Full text available: [pdf\(158.90 KB\)](#)Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Array programming shines in its ability to express computations at a high-level of abstraction, allowing one to manipulate and query whole sets of data at once. This paper presents the OPA model that enhances object-oriented programming with array programming features. The goal of OPA is to determine a minimum set of modifications that must be made to the traditional object model in order to take advantage of the possibilities of array programming. It is based on a minimal extensio ...

**Keywords:** array programming, f-script, high-level language, high-order messages, message pattern, smalltalk

**2 [Mobility: Flexible on-device service object replication with replets](#)**

Dong Zhou, Nayeem Islam, Ali Ismael

 May 2004 **Proceedings of the 13th international conference on World Wide Web WWW '04**

Publisher: ACM Press

Full text available: [pdf\(887.11 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

An increasingly large amount of Web applications employ service objects such as Servlets to generate dynamic and personalized content. Existing caching infrastructures are not well suited for caching such content in mobile environments because of disconnection and weak connection. One possible approach to this problem is to replicate Web-related application logic to client devices. The challenges to this approach are to deal with client devices that exhibit huge divergence in resource ...

**Keywords:** capability, preference, reconfiguration, replication, service, synchronization

**3 [Ellie language definition report](#)**



B. Andersen

November 1990 **ACM SIGPLAN Notices**, Volume 25 Issue 11



Publisher: ACM Press

Full text available: [pdf\(1.68 MB\)](#) Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

This report defines the parallel object-oriented language Ellie in details by means of informal definitions and examples. The syntax of Ellie is defined by a grammar. Ellie has been designed as a part of my Ph.D. thesis. The goals of Ellie are to achieve machine independent parallel programming and great language flexibility. Machine independent parallel programming is achieved by allowing the programmer to have a huge number of small processes for a virtual distributed memory parallel computer. ...

#### 4 Ada, C, C++, and Java vs. the Steelman



David A. Wheeler

July 1997 **ACM SIGAda Ada Letters**, Volume XVII Issue 4



Publisher: ACM Press

Full text available: [pdf\(1.57 MB\)](#) Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

This paper compares four computer programming languages (Ada95, C, C++, and Java) with the requirements of "Steelman", the original 1978 requirements document for the Ada computer programming language. This paper provides a view of the capabilities of each of these languages, and should help those trying to understand their technical similarities, differences, and capabilities.

#### 5 Better extensibility through modular syntax



Robert Grimm

June 2006 **ACM SIGPLAN Notices , Proceedings of the 2006 ACM SIGPLAN conference on Programming language design and implementation PLDI '06**, Volume 41



Issue 6

Publisher: ACM Press

Full text available: [pdf\(233.06 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

We explore how to make the benefits of modularity available for syntactic specifications and present *Rats!*, a parser generator for Java that supports easily extensible syntax. Our parser generator builds on recent research on parsing expression grammars (PEGs), which, by being closed under composition, prioritizing choices, supporting unlimited lookahead, and integrating lexing and parsing, offer an attractive alternative to context-free grammars. PEGs are implemented by so-called packrat ...

**Keywords:** extensible syntax, module system, packrat parsing, parser generator, parsing expression grammar

#### 6 Fortran 8X draft



Loren P. Meissner

December 1989 **ACM SIGPLAN Fortran Forum**, Volume 8 Issue 4



Publisher: ACM Press

Full text available: [pdf\(21.36 MB\)](#) Additional Information: [full citation](#), [abstract](#), [index terms](#)

**Standard Programming Language Fortran.** This standard specifies the form and establishes the interpretation of programs expressed in the Fortran language. It consists of the specification of the language Fortran. No subsets are specified in this standard. The previous standard, commonly known as "FORTRAN 77", is entirely contained within this standard, known as "Fortran 8x". Therefore, any standard-conforming FORTRAN 77 program is standard conforming under this standard. New features can b ...

#### 7 Load-tolerant differentiation with active queue management

 U. Bodin, O. Schelen, S. Pink  
July 2000 **ACM SIGCOMM Computer Communication Review**, Volume 30 Issue 3

**Publisher:** ACM Press

Full text available:  pdf(1.45 MB) Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

Current work in the IETF aims at providing service differentiation on the Internet. One proposal is to provide loss differentiation by assigning levels of drop precedence to IP packets. In this paper, we evaluate the active queue management (AQM) mechanisms RED In and Out (RIO) and Weighted RED (WRED) in providing levels of drop precedence under different loads. For low drop precedence traffic, FIO and WRED can be configured to offer sheltering (i.e., low drop precedence traffic is protected ...)

## **8 Technical Correspondence: Definition of open language**

 Boris Sunik  
February 2001 **ACM SIGPLAN Notices**, Volume 36 Issue 2

**Publisher:** ACM Press

Full text available:  pdf(712.75 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#)

Open languages are the new class of formal languages proposed by the author. A language of this class combines the grammar of an object-oriented programming language with the universality of a natural language. The open language allows the creation of a monolingual communication environment where it can be used in a wide range of communication processes requiring explicit information, i.e. programming, documenting of software and hardware, specification of various standards, tasks' setting, pres ...

**Keywords:** C++, language, object-oriented programming, programming language

## **9 An executable language definition**

 W. M. Waite  
February 1993 **ACM SIGPLAN Notices**, Volume 28 Issue 2

**Publisher:** ACM Press

Full text available:  pdf(917.03 KB) Additional Information: [full citation](#), [citations](#), [index terms](#)

## **10 Technical contributions: Language design for the Ironman requirement: reference**

 manual  
Mary Shaw, Paul Hilfinger, Wm. A. Wulf  
September 1978 **ACM SIGPLAN Notices**, Volume 13 Issue 9

**Publisher:** ACM Press

Full text available:  pdf(1.05 MB) Additional Information: [full citation](#), [abstract](#), [citations](#)

Tartan is an experiment in language design. The goal was to determine whether a "simple" language could meet substantially all of the Ironman requirement for a common high-order programming language. We undertook this experiment because we believed that all the designs done in the first phase of the DOD effort were too large and too complex. We saw that complexity as a serious failure of the designs: excess complexity in a programming language can interfere with its use, even to the extent that a ...

## **11 Technical correspondence: Assessment of the Java programming language for use in high integrity systems**

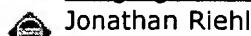
 Jagun Kwon, Andy Wellings, Steve King  
April 2003 **ACM SIGPLAN Notices**, Volume 38 Issue 4

**Publisher:** ACM Press

Full text available: [pdf\(319.62 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#)

This paper sets a goal of investigating the use of Java in the development of high integrity systems. Based on previous studies, guidelines, and standards, we develop 23 criteria that are used for the following assessment of Java. A summary of the assessment is provided before we go on to review a few existing subsets of the language.

**12 Staging: Assimilating MetaBorg:: embedding language tools in languages** □



Jonathan Riehl

October 2006 **Proceedings of the 5th international conference on Generative programming and component engineering GPCE '06**

Publisher: ACM Press

Full text available: [pdf\(174.54 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

The MetaBorg usage pattern allows concrete syntax to be associated with application programmer interfaces (API's). Once a concrete syntax is defined, library writers use the Stratego language to write transformations from the concrete syntax to API data and calls in the host language. The result is a compile time translator from the combined host and domain languages to the host language. This translator is not programmable at compile time, and little or none of the infrastructure can be leverag ...

**Keywords:** MetaBorg, SDF, concrete syntax macros, extensible syntax, self application, staged multi--language programming, stratego

**13 Report on the 5th IFIP international workshop on quality of service (IWQOS'97)** □



Oguz Angin, Andrew T. Campbell, Lai-Tee Cheok, Raymond R-F Liao, Koon-Seng Lim, Klara Nahrstedt

July 1997 **ACM SIGCOMM Computer Communication Review**, Volume 27 Issue 3

Publisher: ACM Press

Full text available: [pdf\(1.86 MB\)](#) Additional Information: [full citation](#), [abstract](#), [index terms](#)

This paper presents a summary of the fifth International Workshop on Quality of Service (IWQOS) which was held at Columbia University in May 1997. The goal of this three-day meeting was to foster interaction between researchers active in the area of Quality of Service(QOS) research, to reflect on past experiences and lessons learnt, and to discuss future QOS challenges. To reflect this goal, this year's workshop included a hot program made up of (i) a keynote address on "Programming Telecommunic ...

**14 Answering English questions by computer: a survey** □



R. F. Simmons

January 1965 **Communications of the ACM**, Volume 8 Issue 1

Publisher: ACM Press

Full text available: [pdf\(2.79 MB\)](#) Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

**15 Session III: Type-safe, self inspecting code** □



Arthur I. Baars, S. Doaitse Swierstra

September 2004 **Proceedings of the 2004 ACM SIGPLAN workshop on Haskell Haskell '04**

Publisher: ACM Press

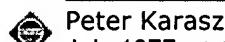
Full text available: [pdf\(135.76 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

We present techniques for representing typed abstract syntax trees in the presence of observable recursive structures. The need for this arose from the desire to cope with left-

recursion in combinator based parsers. The techniques employed can be used in a much wider setting however, since it enables the inspection and transformation of any program structure, which contains internal references. The hard part of the work is to perform such analyses and transformations in a setting in which the Ha ...

**Keywords:** compilers, domain specific languages, left-recursion, top-down parsing

**16 An intermediate systems language for the PDP-11**



Peter Karasz

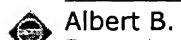
July 1977 **ACM SIGPLAN Notices**, Volume 12 Issue 7

**Publisher:** ACM Press

Full text available: [pdf\(1.23 MB\)](#) Additional Information: [full citation](#), [abstract](#)

In this paper we describe Version 2 of Intermediate Systems Language (ISL/V2), a general purpose software implementation language developed at the B. F. Goodrich Company. The particular implementation we discuss here is for the PDP-11 family of minicomputers that use the 11/40 instruction set. ISL has a FORTRAN like syntax with a number of added features to make it a versatile software development tool. It is the implementation language of IDMS-11, a CODASYL-type Data Base Management System (DBMS ...

**17 Transient analysis of a store-and-forward computer-communications network**



Albert B. Garcia, Wade H. Shaw

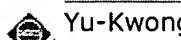
December 1986 **Proceedings of the 18th conference on Winter simulation WSC '86**

**Publisher:** ACM Press

Full text available: [pdf\(706.98 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

This paper presents the results of a study of transient behavior in a store-and-forward, computer-communications network. The purpose of this paper is to discuss the nature of network performance during severe loading of a network previously operating at steady state conditions. This knowledge is critical since networks are generally designed for a maximum steady state message load. When a transient situation develops, the network may not perform as originally intended and may produce excess ...

**18 Static scheduling algorithms for allocating directed task graphs to multiprocessors**



Yu-Kwong Kwok, Ishfaq Ahmad

December 1999 **ACM Computing Surveys (CSUR)**, Volume 31 Issue 4

**Publisher:** ACM Press

Full text available: [pdf\(723.58 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Static scheduling of a program represented by a directed task graph on a multiprocessor system to minimize the program completion time is a well-known problem in parallel processing. Since finding an optimal schedule is an NP-complete problem in general, researchers have resorted to devising efficient heuristics. A plethora of heuristics have been proposed based on a wide spectrum of techniques, including branch-and-bound, integer-programming, searching, graph-theory, randomization, genetic ...

**Keywords:** DAG, automatic parallelization, multiprocessors, parallel processing, software tools, static scheduling, task graphs

**19 The MAD definition facility**

Bruce W. Arden, Bernard A. Galler, Robert M. Graham

 August 1969 **Communications of the ACM**, Volume 12 Issue 8

Publisher: ACM Press

Full text available:  pdf(1.04 MB)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

One of the first definition facilities for higher level languages is described. Users of the language can define new operators and/or data types into the MAD language, so that their use appears as if they were predefined. Information is given on how one writes definitions, as well as on much of the motivation behind the form in which definitions are written. Some conclusions are drawn about future definitional facilities.

**Keywords:** MAD, definitions, higher level language, macros, operators

20 A modest proposal: C++ resyntaxed 

 Ben Werther, Damian Conway

November 1996 **ACM SIGPLAN Notices**, Volume 31 Issue 11

Publisher: ACM Press

Full text available:  pdf(745.78 KB) Additional Information: [full citation](#), [abstract](#), [index terms](#)

We describe an alternative syntactic binding for C++. This new binding includes a completely redesigned declaration/definition syntax for types, functions and objects, a simplified template syntax, and changes to several problematic operators and control structures. The resulting syntax is LALR(1) parsable and provides better consistency in the specification of similar constructs, better syntactic differentiation of dissimilar constructs, and greater overall readability of code.

Results 1 - 20 of 51

Result page: [1](#) [2](#) [3](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2007 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)